

MULTI-DIMENSIONAL SYNCHRONIZATION FOR RHYTHMIC SONIFICATION

Jeffrey E. Boyd and Andrew Godbout

Department of Computer Science
University of Calgary
Calgary, AB, Canada T2N 1N4
boyd@cpsc.ucalgary.ca, agodabout@ucalgary.ca

ABSTRACT

Human locomotion is fundamentally periodic, so when sonifying gait, it is desirable to exploit this periodicity to produce rhythmic sonification synchronized to the motion. To achieve this rhythmic sonification, some mechanism is required to synchronize an oscillator to the period of the motion. This paper presents a method to synchronize to multidimensional signals like those produced by a motion capture system. Using a subset of the joint-angle signals produced by motion capture, the method estimates the phase of a periodic, multidimensional model to match data observed from a moving subject. It does this using an optimization algorithm applied to a suitable objective function. We demonstrate the synchronization with data from a publicly available motion capture database, producing sonifications of drum beats synchronized to footfalls of subjects. The method is robust and shares some common features of phase-locked loops used for synchronizing one-dimensional sinusoidal signals. We foresee applications to sonification for athletics and clinical treatment of gait disorders.

1. INTRODUCTION

Human locomotion is, by necessity, periodic in nature [1]. Walking, jogging, running, rowing, and skating are common examples in which periodic repetition of motions move a person. We seek to use sonification to assist the training of athletes and in the clinical treatment of gait disorders. Given the periodic nature of locomotion, it then seems natural (possibly even required) to exploit this periodicity in sonification. This requires that the sonification system operate synchronously with the motion, resulting in *rhythmic sonification*.

Figure 1 illustrates the concept of rhythmic sonification. A phase signal, $\phi(t)$ (normalized such that $0 \leq \phi < 1$) provides a temporal base indicating where a subject is in the cycle of a walking stride (or other periodic motion). As $\phi(t)$ passes a phase threshold, ϕ_T , it triggers a sonic event. For example, one can select ϕ_T to correspond to the right footfall resulting in a sound that occurs synchronously with the rhythm of the walker. $\phi(t)$ is the foundation upon which one builds rhythmic sonification – once $\phi(t)$ is established, a plethora of options for rhythmic sonification becomes available.

Godbout and Boyd [2] give an example of rhythmic sonification in speed skating. They measure the ankle angle of a skater over time and synchronize to a model to generate a $\phi(t)$, and use that to provide rhythmic audio feedback to the skater. However, ankle angle measured over time is a one-dimensional signal. In contrast, motion capture systems generate many channels of data that we may wish to synchronize to. For example, the skeletal

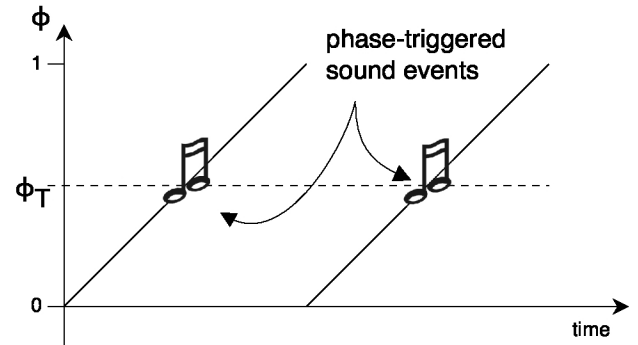


Figure 1: Phase-triggered sound events. Phase cycles from zero through one over the course of one period of the gait (or other periodic motion). As the phase passes a threshold, ϕ_T , it triggers a sound event to give rhythmic sonification synchronized to the motion.

poses measured by a Vicon [3] system in the CMU Motion Capture Database [4] contains 62 channels of data. Boyd and Sadikali [5] describe a rhythmic sonification system using multiple channels of pixel data, but each channel is synchronized separately.

In this paper, we present a novel synchronization method to produce a *synchronized time base from multi-dimensional motion capture data*. Using multidimensional data not only provides a more reliable synchronization, but opens the doors to rhythmic sonification with numerous sensors beyond motion capture system, e.g., multi-axis accelerometers and gyros. We demonstrate our method with examples of walking and running motion capture data. The method provides a reliable time base along with a measure indicating the quality of synchronization at any point in time.

2. BACKGROUND

The synchronization of periodic events is a common phenomenon [6]. Synchronization shows up in electrical and mechanical systems, mathematics, psychology, and biological systems.

Phase-locked loops (PLL) [7] are a well known mechanism for synchronizing sinusoidal signals. PLLs are essentially feedback control systems that adjust the frequency of an internal sinusoidal oscillator to synchronize to an external oscillation. They are widely used in communications systems. Ijspeert et al. [8] and Pongas et al. [9] give examples of multi-dimensional synchronization in robotics. They measure and model periodic motions to

build control systems that allow robots to duplicate these periodic actions.

While PLLs synchronize a single oscillator, Strogatz et al. [10, 11, 12] examined the mutual synchronization of multiple oscillators. Inspired by natural phenomena such as the synchronization of fireflies, they established the regions within the space of coupling parameters that result in synchronization.

The importance of synchronization has been observed in the psychology literature. For example, Bertenthal and Pinto [13] use moving light displays to show the importance of phase locking in the perception of human gaits. When phase locking of the lights is perturbed, observers do not readily perceive a gait.

In biological systems, McGeer [1] showed that periodicity in human locomotion is an inevitable and natural consequence of the structure of the human body – gait is a limit cycle arising from body mechanics. Glass [14] examines possible mechanism for synchronization in biological structures. Cariani [15, 16, 17, 18] describes temporal coding mechanisms for perception of sound.

The message is clear – where moving people are concerned, synchronization is important. Therefore, when one seeks to sonify human motion, synchronizing to the motion is important, perhaps even necessary and we see examples in the work of Staum [19], Hamburg and Clair [20], Godbout and Boyd [2], and Boyd and Sadikali [5].

3. SYNCHRONIZATION BY OPTIMIZATION

Let $\mathbf{y}(k) = [y_1(k) \dots y_{n_c}(k)]^T$ be a vector of measurements of a periodic n_c -dimensional signal at time interval k . For example, Figure 2(a) shows an example walking gait from the CMU Motion Capture Database [4], $n_c = 4$. Note that although the full data set has 62 channels, we use only a subset for the synchronization. We choose the subset to contain those channels we expect will be best for synchronization. For example, hand and wrist movements are likely to confound the process, while McGeer [1] suggests that leg motion must be periodic. Therefore, we use the left and right femur and tibia, and take only the channels corresponding to motion in the sagittal plane (x -axis rotation as denoted in the database). This corresponds to rotation about the hip and knee joints. In the remaining discussion, we assume that each channel of \mathbf{y} is zero-mean, or has been preprocessed (with a high-pass filter) so that it is zero-mean. Our multidimensional synchronization process follows these steps.

1. Build a multidimensional periodic model of the motion we wish to synchronize to. This needs to be done only once for any type of motion (e.g., walking or running).
2. For an unknown signal, match the signal to the model at any point in time to estimate the phase.

The following subsections describe these steps in detail.

3.1. The Model

Let $\mathbf{y}_e(k)$ be an exemplar signal with n_s samples for the motion we wish to synchronize with. It must contain at least one full period of the motion. Our goal is to build a model function, $f(\phi(k))$, that approximates $\mathbf{y}_e(k)$. Equivalently, we want n_c models such that $f_i(\phi(k)) \approx y_i(k)$ for $1 \leq i \leq n_c$.

Taking inspiration from Ijspeert et al. [8], we build f_i from a linearly weighted combination of circular Gaussian basis func-

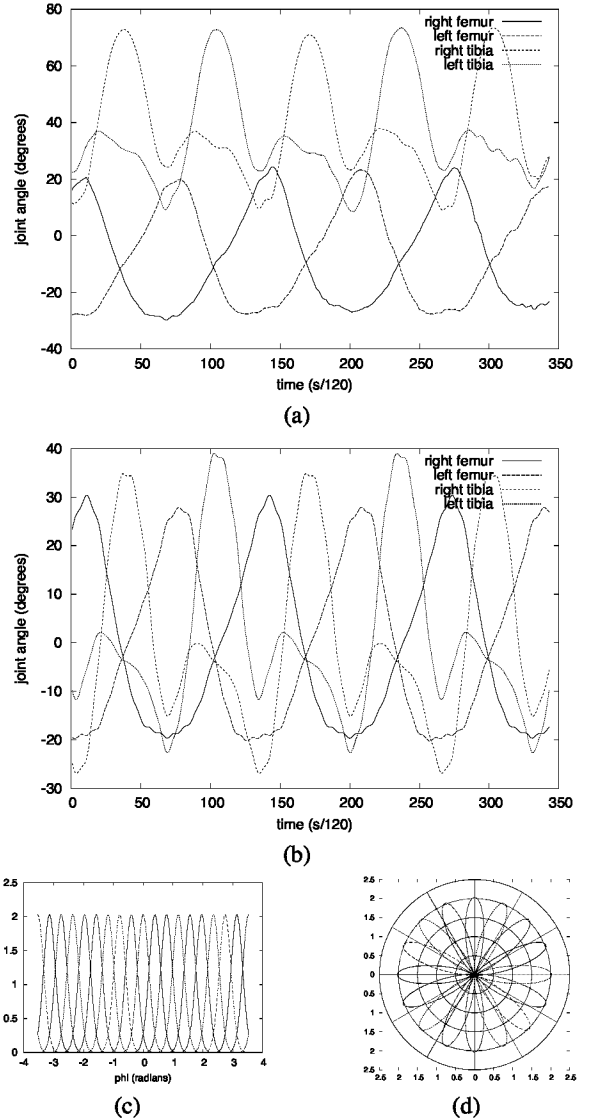


Figure 2: Signals used in the construction of a multidimensional periodic model: (a) the x -axis rotation of the left and right femur and tibia, (b) the model obtained for $n_c = 4$ and $n_m = 16$, (c) the periodic basis functions for $n_m = 16$, and (d) the same basis functions plotted in polar coordinates.

tions. That is:

$$f_i(\phi) = \sum_{j=1}^{n_m} w_{ij} g(\phi; \mu_j, \sigma), \quad (1)$$

where n_m is the number of Gaussian basis functions in our model, w_{ij} is weight of the j^{th} Gaussian for the i^{th} channel, and

$$g(\phi; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(\phi-\mu)/2\sigma^2}, \quad (2)$$

is the Gaussian probability density function with center μ and standard deviation σ .

We select the μ_j such that the Gaussian bases are uniformly distributed between $-\pi$ and π , and separated by 2σ , i.e., $\sigma = \pi/n_m$. Larger values of n_m give a basis set that models \mathbf{y} in more (high-frequency) detail, and lower values for n_m lead to a model of \mathbf{y} that is smoother and has less high-frequency detail. Figure 2(c) and (d) show the basis functions for $n_m = 16$.

The set of w_{ij} for $1 \leq i \leq n_c$ and $1 \leq j \leq n_m$ defines our model function. To obtain the w_{ij} from $y_i(k)$, for $1 \leq k \leq n_s$, we build the following system of equations,

$$\begin{bmatrix} g(\phi(1); \mu_1, \sigma) & \dots & g(\phi(1); \mu_{n_m}, \sigma) \\ \vdots & \ddots & \vdots \\ g(\phi(n_s); \mu_1, \sigma) & \dots & g(\phi(n_s); \mu_{n_m}, \sigma) \end{bmatrix} \begin{bmatrix} w_{i,1} \\ \vdots \\ w_{i,n_m} \end{bmatrix} = \begin{bmatrix} y_{e_i}(1) \\ \vdots \\ y_{e_i}(n_s) \end{bmatrix}, \quad (3)$$

and solve using least squares. To get $\phi(k)$, we arbitrarily select an easily identified point in \mathbf{y}_e and use that to establish a phase reference such that ϕ ramps from zero to 2π over each period of \mathbf{y} . For what follows, we use the first two zero crossings of the x -axis rotation of the right femur with positive slope. Figure 2(b) shows the model obtained for the exemplar in Figure 2(a) for $n_c = 4$ and $n_m = 16$.

3.2. Synchronization

To synchronize \mathbf{f} with an unknown $\mathbf{y}(k)$ at sample interval k , we maximize an objective function parameterized by phase. We begin with the following:

$$E_1(\phi) = \mathbf{f}(\phi) \otimes_{\text{norm}} \mathbf{y}(k), \quad (4)$$

where \otimes_{norm} denotes normalized cross-correlation.

Maximizing E_1 works well to estimate phase, but often the phase estimates deviate because the subject is not exactly like the exemplar. To *smooth out* the phase estimates, we introduce a second term to our objective function to favour solutions with a constantly increasing phase:

$$E_2(\phi) = \left(\frac{\phi - (\hat{\phi}(k-1) + \Delta\phi)}{2\pi} \right)^2, \quad (5)$$

where $\hat{\phi}(k-1)$ is the phase estimate for the previous sample of \mathbf{y} , and $\Delta\phi$ is the expected phase change between samples based on a typical walking cadence. Minimizing E_2 produces a phase ramp that corresponds exactly to the $\Delta\phi$. We combine E_1 and E_2 to get the following objective function,

$$E(\phi) = E_1(\phi) - \lambda E_2(\phi), \quad (6)$$

where λ is a regularization parameter. When, λ is small, the estimated phase depends primarily on a matching data to the model, and when λ is large, the estimated phase reflects only the cadence defined by $\Delta\phi$, i.e., a period of

$$\frac{2\pi T}{\Delta\phi}, \quad (7)$$

Where T is the sample period. To estimate the phase we compute

$$\hat{\phi} = \underset{\phi}{\operatorname{argmax}} E(\phi), \text{ and} \quad (8)$$

$$E_{\max} = \max_{\phi} E(\phi), \quad (9)$$

where $\hat{\phi}$ is our phase estimate and E_{\max} is a measure of quality of match between signal and model.

As might be expected, $E(\phi)$ is periodic itself, and some care is needed to perform the optimization in the previous equation. We developed the following algorithm to compute $\hat{\phi}$.

1. Compute E on the n_m centers of the Gaussian basis functions, i.e., evaluate $E(\mu_1)$ through $E(\mu_{n_m})$.
2. Find the maximum value of E , $E(\mu_{j_{\max}})$ among the samples in step 1.
3. Interpolate to find the position of the maximum among the samples $E(\mu_{j_{\max}-1})$, $E(\mu_{j_{\max}})$, and $E(\mu_{j_{\max}+1})$.

To interpolate between samples, we use Nishihara's [21] *sub-pixel interpolation* method illustrated in Figure 3. Three adjacent, uniformly spaced samples centered at the origin, $x = -1, 0, 1$, bracket a maximum of $f(x)$. The three points define a parabola. Some basic calculus reveals that the position of the maximum, x_m is at

$$x_m = \frac{-b}{2a}, \quad (10)$$

where

$$a = \frac{1}{2}(f(1) + f(-1)) - f(0), \text{ and} \quad (11)$$

$$b = \frac{1}{2}(f(1) - f(-1)). \quad (12)$$

The maximum value estimated by interpolation is

$$f(x_m) = ax_m^2 + bx_m + c, \quad (13)$$

where $c = f(0)$. To find $\hat{\phi}$, and E_{\max} , set

$$f(-1) = E(\mu_{j_{\max}-1}) \quad (14)$$

$$f(0) = E(\mu_{j_{\max}}), \text{ and} \quad (15)$$

$$f(1) = E(\mu_{j_{\max}+1}), \quad (16)$$

interpolate to find x_m and $f(x_m)$, then set

$$\hat{\phi} = \mu_{j_{\max}} + x_m \frac{\pi}{n_m}, \text{ and} \quad (17)$$

$$E_{\max} = f(x_m). \quad (18)$$

4. IMPLEMENTATION AND TESTING

4.1. General

We tested our method using the CMU Motion Capture Database [4]. The database contains motion capture data for multiple subjects performing different activities over multiple trials. The motion capture data is sampled at 120Hz, and is available with raw video of trials, video renderings of the data, and various software tools. Of the activities available in the database, we tested on the complete selection of walking and running examples.

We implemented the method in *Octave* [22], an open-source Matlab variant, then later implemented the optimization algorithm for phase matching in Pure Data [23]. In all cases, we computed the model coefficients, w_{ij} , with Octave since this needs to be done only once, prior to any sonification.

It is necessary to manually choose the exemplar from which the model is built. In the examples here, we chose a single trial for

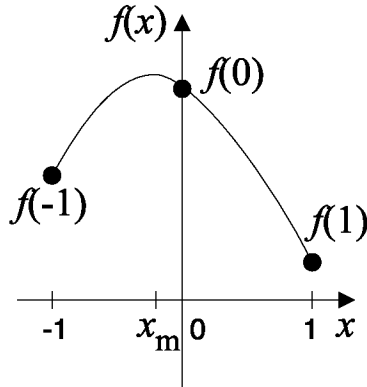


Figure 3: Nishihara sub-pixel interpolation method to find the maximum of a parabola that fits three adjacent samples with uniform spacing.

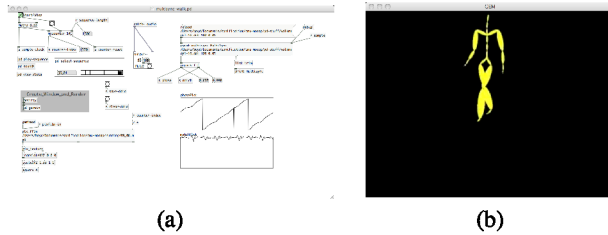


Figure 4: Screenshots from Pure Data sonification patch demonstrating synchronous sonification of multidimensional data: (a) patch, and (b) synchronized video.

each of walking and running with the requirement that the exemplar sequence could contain only the activity of interest, and had to have at least two positive-going zero-crossings in the right femur x -axis rotation. The zero-crossings ensured that we could establish $\phi(k)$ correctly. While it would be possible to combine multiple subjects and trials when computing the model coefficients, it turned out not to be necessary as our results show – it seems one person’s gait is similar enough to others to establish a time base. For all the examples here, walking and running alike, we used $\Delta\phi = 0.05\text{radians}$, which corresponds to a gait period of 1.05s. Also, for all examples, we used $\lambda = 100$.

Our sonification is simple, but sufficient to verify that we have a correct time base for other more complex sonifications. In general, once the time base is correct, timing sound events is simple. With that in mind, our sonification consists of two drum taps per gait period with the phase triggers set to correspond to the left and right footfalls. When viewing the rendered motion capture video with the sonification, it is simple to verify that the drum beats are occurring at the correct time and that the time base is correct. We normalized phases in the range $[-\pi \dots \pi]$ to $[0 \dots 1]$. In this case, footfalls happen at approximately $\phi_T = 0.25$ and $\phi_T = 0.75$. Figure 4 shows screenshots from the Pure Data patch in operation.

4.2. Walking

Figure 5 shows plots of $\hat{\phi}$ and E_{\max} for four representative walking sequences. In all examples we tried, drum beats occurred coin-

cidentally with footfalls in all cases where the subject was walking with a normal stride. As expected, the synchronization only fails when the subject is walking backwards or otherwise not walking normally. In these cases, the subject has deviated too far from our model gait for synchronization to occur.

Figure 5(a) shows plots for our walk training subject, i.e., it shows the model synchronizing to itself – a strawman test. The second term of Equation 6 starts with an arbitrary $\hat{\phi}$ and takes a few samples to converge to the correct phase. After this convergence, the phase is synchronized correctly. The longest convergence period we observed was approximately 75% of a gait cycle, and most often the convergence occurs in half a cycle or less. Note that the values of E_{\max} are low during the convergence interval. So although the system has not converged, it has a numerical indicator that the phase estimate is not good. This example also has a period of 1.1s, which happens to correspond closely to the natural period for $\Delta\phi = 0.05\text{radians}$.

Figure 5(b) shows results for a similar trial, but with a different subject. This subject has a much slower stride, with a period of 1.6s. Although this is significantly different than the natural period for $\Delta\phi = 0.05\text{radians}$, the system correctly locks to the phase of the walker while the second term of Equation 6 smooths the phase estimates.

Figure 5(c) corresponds to a sequence in which the subject walks for a few paces, stops, turns around, and walks a few paces back to their starting position. The synchronization plots clearly show this. In the middle of the plot, there is an interval during which the phase stops ramping and E_{\max} drops which corresponds to the moment when the subject stops and turns. The sonification produces correct footfalls during the normal paces, and a couple of spurious taps as the subject stops and turns.

Walking backwards confounds the synchronization and sonification as shown in Figure 5(d). These plots correspond to part of a sequence where the subject walks backwards for a couple of paces. Clearly the synchronization has failed. The second term of the objective function (Equation 6) drives $\hat{\phi}$ forward in an approximate phase ramp, but waveform is irregular and E_{\max} values are sporadically low indicating a poor match. We did try synchronizing to this sequence with the second term of Equation 6 removed, i.e., $\lambda = 0$. In this case we do see a downward phase ramp as one might expect, but the cost is in a noisier phase estimate throughout the entire sequence.

4.3. Running

Figure 6 shows plots of $\hat{\phi}$ and E_{\max} for four representative running sequences. As was the case with the walking examples, the drum beats occurred simultaneously with footfalls during normal running. Most of the running sequences are by necessity shorter – the higher speed means the subject is in the field of view of the motion capture system for a shorter period of time, unless they alter their gait to change direction.

Figure 6(a) shows synchronization with the same subject used for our running model, but for a different trial. Synchronization is comparable to what we observed for walking. Figure 6(b) shows a sequence for a different subject, again exhibiting excellent synchronization. It is worth noting that although the stride frequencies for these are significantly faster than the natural frequency for $\Delta\phi = 0.05\text{radians}$ (periods of 0.68s and 0.78s versus 1.05s), our system still synchronizes well.

Figure 6(c) corresponds to a sequence in which the subject

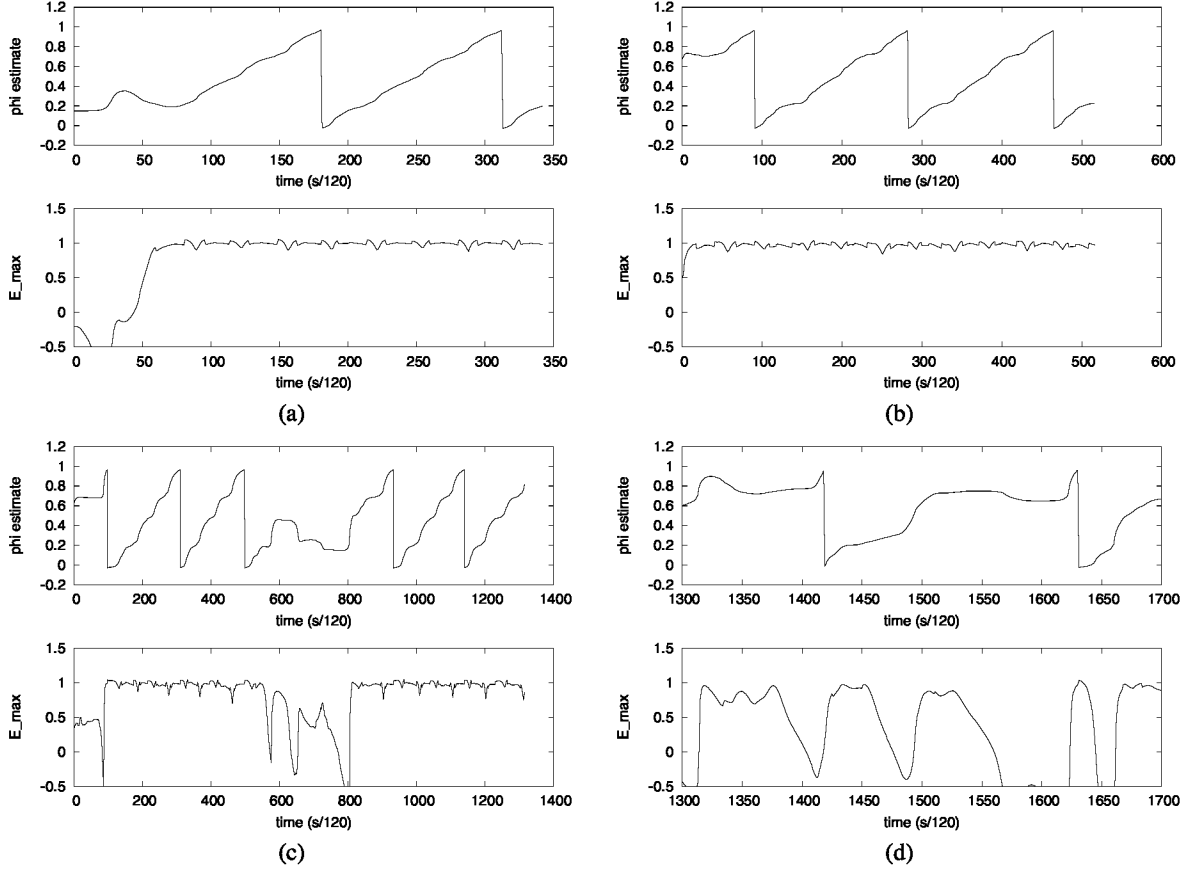


Figure 5: Walking synchronization results: (a) model sequence, (b) a typical walking sequence, (c) subject stopping and turning, and (d) subject walking backward. In all examples, the upper plot shows the phase estimate, $\hat{\phi}$, and the lower plot shows E_{\max} .

runs, comes to a stop, and with a hop changes direction. One can see when the hop occurs where the phase ramp is distorted near the middle of the sequence, and where the sporadic drops in E_{\max} occur. Again, it was in these sorts of variations from a normal running gait where the sonification of footfalls becomes erratic.

Figure 6(d) shows plots for a longer running sequence in which the subject runs around the field of view in a box pattern, turning at the corners. The effects of this pattern are clear in the plots. One can see the dips in E_{\max} at the corners, and also some distortion in the phase ramps as the subject alters the gait to accommodate the corner.

5. DISCUSSION

As a way to understand the synchronization method presented here, we can compare to PLLs. Figure 7 shows the elements of a PLL [7]. The phase comparator and the (low-pass) loop filter together compare input oscillations to the oscillations of an internal oscillator, the *voltage controlled oscillator* (VCO). The transfer function of the VCO, shown in Figure 7(b) relates the frequency of the internal oscillator to its natural frequency, ω_0 , and the difference between internal and external signals. It is not meaningful to compare two one-dimensional signals instantaneously, leading to the requirement to have a low-pass filter that effectively integrates

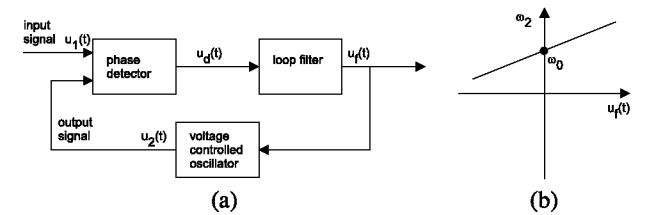


Figure 7: A basic phase-locked loop: (a) block diagram, and (b) the transfer function of the voltage controlled oscillator.

phase comparisons over time.

In their synchronization system for speed skating, Godbout and Boyd [2] also integrate a comparison over time when they compute the normalized cross-correlation over a window of one period. They have no equivalent to the VCO, relying instead on a brute-force search over frequency space for every sample.

In the system presented here, we are getting close to a multi-dimensional PLL for arbitrary wave forms. The E_1 term in Equation 6 compares an incoming multidimensional signal to the internal multidimensional oscillator in our model. The need for the low-pass filter is obviated by the multidimensional signal – we integrate over dimensions instead. This allows us to get an instan-

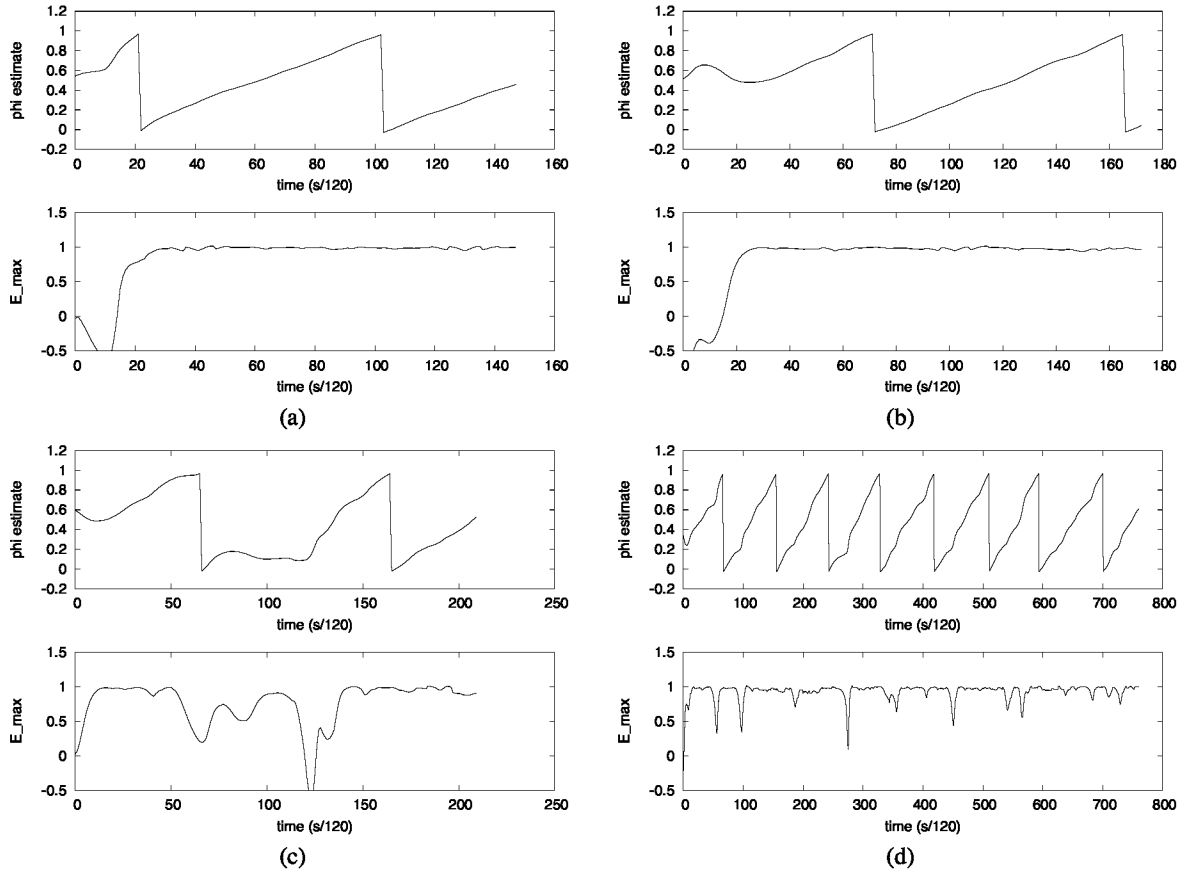


Figure 6: Running synchronization results: (a) the model sequence, (b) a typical running sequence, (c) hop and change of direction, and (d) running around a square. In all examples, the upper plot shows the phase estimate, $\hat{\phi}$, and the lower plot shows E_{\max} .

taneous comparison that is not possible with one-dimensional signals. Further more, the E_2 term in Equation 6 is equivalent to the VCO. It ramps at a natural frequency defined by $\Delta\phi$ but responds to the external signal when combined with E_1 . Our system is not precisely equivalent to a PLL though – it lacks feedback to track the incoming signal, relying on an optimization for each sample interval.

It is important to note that although we synchronize with just four channels of the motion capture data, once we are synchronized, we can rhythmically sonify any and all channels of the data. We see potential here because:

- our method opens the door to real-time rhythmic sonification for athletics and clinical applications, and
- motion capture is getting cheaper (consider the MicroSoft Kinect) which will lower the cost requirements for using this type of sonification.

6. CONCLUSIONS

We have presented a method of synchronization applicable to periodic, multidimensional signals like those produced by motion capture systems acquiring data from locomotion. The system features key elements of PLLs, an established method for synchronizing internal oscillators to incoming sinusoids. Once this synchronization

is established, it provides the temporal basis for rhythmic sonification.

7. REFERENCES

- [1] T. McGeer, “Passive walking with knees,” in *IEEE International conference on Robotics and Automation*, 1990, pp. 1640–1645.
- [2] A. Godbout and J. E. Boyd, “Corrective sonic feedback for speed skating: a case study,” in *International Conference on Auditory Display*, Washington, DC, June 2010, pp. 23–30.
- [3] “Motion capture systems from vicon,” Retrieved January 23, 2012, from <http://www.vicon.com>.
- [4] “Cmu graphics lab motion capture database,” <http://mocap.cs.cmu.edu/>, created with funding from NSF EIA-0196217.
- [5] J. E. Boyd and A. Sadikali, “Rhythmic gait signatures from video without motion capture,” in *International Conference on Auditory Display*, Washington, DC, June 2010, pp. 187–191.
- [6] S. Strogatz, *Sync: The Emerging Science of Spontaneous Order*. New York: Theia Books, 2003.

- [7] R. E. Best, *Phase-locked loops design, simulation and applications*. New York: McGraw-Hill, 1999.
- [8] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Learning rhythmic movements by demonstration using nonlinear oscillators," in *International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland, October 2002, pp. 958–963.
- [9] D. Pongas, A. Billard, and S. Schaal, "Rapid synchronization and accurate phase-locking of rhythmic motor primitives," in *International Conference on Intelligent Robots and Systems*, Edmonton, AB, Canada, August 2005, pp. 2911–2916.
- [10] P. C. Matthews and S. H. Strogatz, "Phase diagram for the collective behavior of limit-cycle oscillators," *Physical Review Letters*, vol. 65, no. 14, pp. 1701–1704, October 1990.
- [11] R. E. Mirollo and S. H. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM Journal of Applied Mathematics*, vol. 50, no. 6, pp. 1645–1662, November 1990.
- [12] S. H. Strogatz, R. E. Mirollo, and P. C. Matthews, "Coupled nonlinear oscillators below the synchronization threshold: relaxation by generalized Landau damping," *Physical Review Letters*, vol. 68, no. 18, pp. 2730–2733, May 1992.
- [13] B. I. Bertenthal and J. Pinto, "Complementary processes in the perception and production of human movements," in *A Dynamic Systems Approach to Development: Applications*, L. B. Smith and E. Thelen, Eds. Cambridge, MA: MIT Press, 1993, pp. 209–239.
- [14] L. Glass, *Nonlinear Dynamics in Physiology and Medicine*, ser. Interdisciplinary Applied Mathematics. Springer, 2003, ch. Resetting and entraining biological rhythms, pp. 123–148.
- [15] P. Cariani, "Temporal coding of periodicity pitch in the auditory system: an overview," *Neural Plasticity*, vol. 6, no. 4, pp. 147–172, 1999.
- [16] — —, "Temporal coding of sensory information in the brain," *Acoustical Science and Technology*, vol. 22, no. 2, pp. 77–84, 2001.
- [17] — —, "Temporal codes, timing nets, and music perception," *Journal of New Music Research*, vol. 30, no. 2, pp. 107–135, 2002.
- [18] — —, "Temporal codes and computations for sensory representation and scene analysis," *IEEE Transactions on Neural Networks*, vol. 15, no. 5, pp. 1100–1111, 2004.
- [19] M. J. Staum, "Music and rhythmic stimuli in the rehabilitation of gait disorders," *Journal of music therapy*, vol. 20, no. 2, pp. 69–87, 1983.
- [20] J. Hamburg and A. A. Clair, "The effects of a movement with music program on measures of balance and gait speed in healthy older adults," *Journal of Music Therapy*, pp. 212–226, 2003.
- [21] H. K. Nishihara, "Prism: A practical real-time imaging stereo matcher," MIT AI Lab, Tech. Rep. AI Memo 780, 1984.
- [22] "Octave," Retrieved February 8, 2012, from <http://www.gnu.org/software/octave/>.
- [23] "Pure data," Retrieved February 8, 2012, from <http://puredata.info/>.